

Amendments to the claims:

This listing of claims replaces all prior versions and listings of claims in the application.

Listing of claims

1. (Currently amended) A processor-based method for determining processor usage by a thread, comprising:

determining elapsed times between execution points of the thread based on start times and stop times associated with the execution points, wherein each execution point is an instruction in program code;

determining for each elapsed time whether the thread was idle during the elapsed time by comparison of the elapsed time to a first threshold value;

reducing to a selected value each elapsed time during for which the thread was determined to be idle; and

determining a value indicative of processor usage by the thread as a function of the elapsed times.

2. (Currently amended) The method of claim 1, wherein the ~~reducing to the selected value each elapsed time during which the thread was idle~~ comprises:

reducing to the selected value each elapsed time that exceeds a second threshold value, wherein the second threshold value is greater than the first threshold value; and

reducing to the selected value each elapsed time that exceeds the first threshold value and does not exceed the second threshold value if a condition was detected that indicates the thread was idle during each elapsed time.

3. (Original) The method of claim 2, wherein the condition comprises a processor interrupt.

4. (Original) The method of claim 2, wherein the condition comprises a context switch between threads of execution of a processor.

5. (Original) The method of claim 2, further comprising determining the second threshold value based on a maximum-length instruction path between execution points of the thread.
6. (Original) The method of claim 1, wherein the first threshold value comprises a minimum time required for an operating system kernel running the thread to execute two thread switches.
7. (Original) The method of claim 1, wherein the threads comprise threads running in a Java virtual machine.
8. (Original) The method of claim 1, wherein determining elapsed times comprises determining the elapsed time using a high-resolution clock.
9. (Original) The method of claim 8, wherein the high-resolution clock comprises a CPU clock.
10. (Original) The method of claim 1, wherein the execution points comprise entry points and exit points of functions called by the thread.
11. (Currently amended) A computer-readable storage medium configured with instructions for causing a processor of a data processing arrangement to perform steps comprising:
 - running a thread of a multi-threaded program;
 - determining elapsed times between execution points of the thread based on start times and stop times associated with the execution points;
 - determining for each elapsed time whether the thread was idle during the elapsed time by comparison of the elapsed time to a first threshold value;
 - reducing to a selected value each elapsed time during for which the thread was determined to be idle; and

determining a value indicative of processor usage by the thread as a function of the elapsed times.

12. (Currently amended) The computer-readable storage medium of claim 11, wherein ~~the~~ reducing to the selected value each elapsed time during which the thread was idle comprises:

reducing to the selected value each elapsed time that exceeds a second threshold value, wherein the second threshold value is greater than the first threshold value; and

reducing to the selected value each elapsed time that exceeds the first threshold value and does not exceed the second threshold value if a condition was detected that indicates the thread was idle during each elapsed time.

13. (Currently amended) The computer-readable storage medium of claim 12, wherein the condition comprises a processor interrupt.

14. (Currently amended) The computer-readable storage medium of claim 12, wherein the condition comprises a context switch between threads of execution of the data processing arrangement.

15. (Currently amended) The computer-readable storage medium of claim 12, wherein the steps further comprise determining the second threshold value based on a maximum-length instruction path between execution points of the thread.

16. (Currently amended) The computer-readable storage medium of claim 11, wherein the first threshold value comprises a minimum time required for an operating system kernel of the data processing arrangement to execute two thread switches.

17. (Currently amended) The computer-readable storage medium ~~method~~ of claim 11, wherein the threads comprise threads running in a Java virtual machine.

18. (Currently amended) The computer-readable storage medium of claim 11, wherein determining elapsed times comprises determining the elapsed time using a high-resolution clock.

19. (Currently amended) The computer-readable storage medium of claim 18, wherein the high-resolution clock comprises a CPU clock.

20. (Currently amended) The computer-readable storage medium of claim 11, wherein the execution points comprise entry points and exit points of functions called by the thread.

21. (Currently amended) An apparatus comprising:

means for running a thread of a multi-threaded program;

means for determining elapsed times between execution points of the thread based on start times and stop times associated with the execution points;

means for determining for each elapsed time whether the thread was idle during the elapsed time by comparison of the elapsed time to a first threshold value;

means for reducing to a selected value each elapsed time during for which the thread was determined to be idle; and

means for determining a value indicative of processor usage by the thread as a function of the elapsed times.